REMARKS

In the Office Action, the Examiner provisionally rejected claims 8 and 18 under the judicially-created doctrine of objectiveness-type double patenting as being unpatentable over claims 4, 12, and 19 of co-pending U.S. Patent Application No. 10/676,374 of Cherdron et al. ("Cherdron"); objected to claims 7, 8, and 11 due to typographical mistakes in claim dependency; objected to claims 21-22 due to grammatical informality; rejected claims 20-22 under 35 U.S.C. § 112, second paragraph, as being indefinite; and rejected claims 1-22 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,964,053 to Ho et al. ("Ho").

By this Amendment, Applicants have amended claims 7, 8, 11, and 20-22. Claims 1-22 are currently pending.

A. Double Patenting Rejections

The Examiner provisionally rejected claims 8 and 18 under the judicially-created doctrine of obviousness-type double patenting over claims 4, 12, and 19 of Cherdron. Applicants respectfully request that the Examiner hold all double-patenting rejections in abeyance until the indication of otherwise allowable subject matter. Upon review of the remarks and arguments made in this paper, should the Examiner believe this application to be in condition for allowance but for the double patenting rejections held in abeyance, Applicants respectfully request that the Examiner contact the undersigned representative to discuss an appropriate resolution.

B. Objections to Claims 7, 8, 11, 21, and 22

In response to the Examiner's objections to claims 7, 8, and 11, Applicants have amended claim 7 to depend from claim 6; claim 8 to depend from claim 7; and claim 11 to depend from claim 10. In response to the Examiner's objections to claims 21 and 22, Applicants have amended claims 21 and 22 to replace the term "delete" with the term "delete command," as suggested by the Examiner. Accordingly, Applicants respectfully request withdrawal of the objections to claims 7, 8, 11, 21, and 22.

C. Rejections of Claims 20-22 Under 35 U.S.C. § 112, second paragraph

Applicants have amended claim 20 to establish sufficient antecedent basis for claims 20-22, as suggested by the Examiner. Accordingly, Applicants respectfully request withdrawal of the § 112 rejections of claims 20-22.

D. Rejections of Claims 1-22 Under 35 U.S.C. § 102(e)

1. Claims 1-9

Applicants respectfully traverse the rejection of claim 1 under 35 U.S.C. § 102(e) as being anticipated by Ho for at least the reason that Ho fails to disclose the combination of steps recited in claim 1. For example, claim 1 recites, among other things, "based on the set of intermediate objects and a code template, generate an API to access the development objects." The Examiner argued that Ho teaches an apparatus to, "based on the set of intermediate objects and a code template, generate an API to access the development objects" because "metamodel languages and descriptor metamodel files used to generate a common Connector metamodel reads on one API." (Office Action, p.6.) Applicants respectfully disagree.

In rejecting claim 1, the Examiner asserted that Rose File 601 of Ho teaches "a first data model in a first language." Id. The Examiner, however, did not clearly indicate which entities disclosed in Ho teach the "development objects." Because the data model (Rose File 601) is being used to structure development objects, it appears from

FIG 6. of Ho that the Examiner meant that the metamodels disclosed in Ho, such as COBOL metamodel, PL/I metamodel, MFS metamodel, and BMS metamodels, teach development objects. The Examiner further asserted that DTD files and XML schemas generated from metamodel files teach "a set of intermediate objects" generated "based on the first data model." (Office Action, p.6.)

Following the Examiner's arguments with respect to the first two claim elements, in order for Ho to teach an apparatus to, "based on the set of intermediate objects and a code template, generate an API to access the development objects," Ho must disclose an apparatus to, based on the DTD files, XML schemas, or Java codes generated from the metamodel files (or the Rose files) and a code template, generate an API to access the metamodels. Nowhere does Ho disclose or suggest such an apparatus.

The Examiner asserted that the connector disclosed in Ho teaches an API to access the development objects. The connector, however, is not an API (Application Programming Interface) to access the development objects. Instead, Ho teaches that the connector is a dynamic, *run-time, interface between platforms*. (Ho, col. 9, lines 26-29.) Furthermore, nowhere does Ho disclose that the connector is generated based on the set of intermediate objects.

The Examiner also asserted that API disclosed in Ho teaches an API to access the development objects. However, the API disclosed in Ho is not an API to access the development objects. Rather, the API of Ho is an API to access the enterprise application that the connector is interfacing with. Furthermore, nowhere does Ho disclose that the API to access the enterprise application is generated based on the set of intermediate objects. FIG. 7 of Ho supports this position. Ho discloses that

Interfaces 705 "are the access points to the applications 703 through which all input and output is connected to the middleware 713." (Ho, col. 11, lines 33-41.) Ho further discloses that Interfaces 705 "are described in terms of the Application Interface Metamodel," and thus are not based on the set of intermediate objects and a code template. Id.

Moreover, FIG. 4 of Ho depicts an IMS OTMA Application Interface Metamodel 411 on top of and connected to Metamodels 421, 423, 425, and 427. But, the IMS OTMA Application Interface Metamodel is not an API to access the Metamodels. Instead, the IMS OTMA Application Interface Metamodel 411 is a IMS OTMA transaction message that consists of Metamodels 421, 423, 425, and 427. (Ho, col. 15, lines 1-20.) The Invocation Metamodel 421 is the OTMA prefix. The IMS Transaction Message Metamodel 423, COBOL Metamodel 425, and C Metamodel are the message segments of the message for input and output requests. Id.

For the reasons set forth above, Ho does not support the rejection of claim 1. In addition, because claims 2-9 depend from claim 1, Ho does not support the rejections of claims 2-9 for at least the same reasons set forth above in connection with claim 1. Accordingly, Applicants respectfully request that the Examiner withdraw the § 102 rejections of claims 1-9 and allow the claims.

2. Claims 10-17

Applicants respectfully traverse the rejection of claim 10 under 35 U.S.C. § 102(e) as being anticipated by Ho for at least the reason that Ho fails to disclose the combination of steps recited in claim 10. For example, claim 10 recites, among other things, "based on the set of intermediate objects and a code template, generate an XML schema used to implement the development objects." Applicants respectfully submit that the phrase "based on the set of intermediate objects and a code template" is a part of the third claim element, and not the second claim element, unlike the way the Examiner divided claim 10.

The Examiner asserted that DTD files, XML schemas, and Java codes of Ho generated from metamodel files teach "a set of intermediate objects" generated "based on the first data model." (Office Action, pp.7-8.) However, the Examiner failed to assert that Ho teaches an apparatus to, "based on the set of intermediate objects and a code template, generate an XML schema used to implement the development objects." Nowhere does Ho disclose an apparatus to, based on the set of intermediate objects (or DTD files, XML schemas, and Java codes, as asserted by the Examiner) and a code template, generate an XML schema used to implement the development objects.

Ho teaches generating XML documents (or XMI instances) based on Java codes generated from metamodel files. (Ho, FIG. 6.) The Examiner asserted that "XMI instance reads on intermediate objects and template leading to constructing a DTD or XML schema." (Office Action, p.10.) Applicants respectfully disagree. A DTD or XML schema defines rules or constraints for XML documents. XML documents can be constructed in compliance with the rules and constraints defined in a DTD or XML schema, but not vice versa. Therefore, the Examiner assertion that "XMI instances reads on intermediate objects and template leading to constructing a DTD or XML schema" cannot be supported.

For the reasons set forth above, Ho does not support the rejection of claim 10. In addition, because claims 11-17 depend from claim 10, Ho does not support the rejections of claims 11-17 for at least the same reasons set forth above in connection with claim 10. Accordingly, Applicants respectfully request that the Examiner withdraw the § 102 rejections of claims 10-17 and allow the claims.

3. Claims 18-22

The Examiner rejected claim 18 for the reasons set forth with respect to claim 1. As explained above, Ho does not support the rejection of claim 1. Furthermore, claim 18 recites, among other things, "use the API to perform operations on a development objects," which is not recited by claim 1. The Office Action is deficient in that the Examiner failed to specify how Ho discloses an apparatus to "use the API to perform operations on a development objects." For this additional reason, Applicants respectfully request that the Examiner withdraw the rejection of claim 18 and allow the claim.

Claims 19-22 depend from claim 18. Therefore, Ho does not support the rejections of claims 19-22 for at least the same reasons set forth above in connection with claim 18. Furthermore, Ho fails to disclose every claim element of the dependent claims. For example, claim 19 recites that "the API comprises an interface layer, a proxy layer, and a state layer." The Examiner asserted that Ho's definitions of an API, a connector, and a stub teach a proxy layer. (Office Action, p.9, citing Ho, col. 9, lines 13-42.) Ho teaches that the stub acts as a local call or a local proxy for the server object. Even if the stub teaches a proxy layer, as asserted by the Examiner, nowhere does Ho disclose that the API comprises of the stub.

Moreover, the Examiner asserted that "SimpleInstanceTD" on FIG. 10B and InstanceTDBase on FIG. 12-13 teach a state layer. Applicants respectfully disagree.

"SimpleInstanceTD" and "InstanceTDBase" classes do not have attributes that store or keep track of states. In addition, even if the classes are capable of maintaining states, FIG. 10B and FIG. 12-13 illustrate the elements of Type Descriptor, not the API. (Ho, col. 16, lines 38-44.) For these additional reasons, Applicants respectfully request that the Examiner withdraw the rejections of claims 19-23 and allow the claims.

In view of the foregoing amendments and remarks, Applicants respectfully request reconsideration and reexamination of this application and the timely allowance of the pending claims.

Please grant any extensions of time required to enter this response and charge any additional required fees to our deposit account 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: January 26, 2007

By:
Jeffrey A. Berkowitz
Reg. No. 36,743